

# DATA DISTRIBUTING METHOD AND MANAGEMENT NODE

## TECHNICAL FIELD

The disclosure relates to the technical field of distributed data management, and in particular to a data distributing method and a management node.

## 5 BACKGROUND

Generally, it may take several hours or more time to process hundreds of Gigabits (GB) of data, e.g. log file, when using a common computer, while it may take hundreds of hours to process dozens of Terabits (TB) of data, which is too long. People adopt a data distributing method to solve the problem; the data distribution includes that a set of all  
10 data is called table, a set of a part of all data is called sub-table, a table may include a number of sub-tables, and a number of sub-tables are distributed to a number of computing nodes, e.g. 100 computing nodes; each computing node is only for computation of a part of all data. Thus, these computing nodes can compute simultaneously, which can reduce the processing time greatly.

15 A data distributing system usually includes a management node and a number of computing nodes; as shown in Fig. 5, the management node and the computing nodes are interconnected through a network. The management node is responsible for distribution of data, scheduling of computing tasks and so on; the management node is always coincident with a data storage management node and a cluster management  
20 node physically, wherein the data management node records the number of sub-tables of a table, the location of sub-tables and other information, and the cluster management node records the addresses of nodes, memory size, CPU capability and other information. The computing node is responsible for storing sub-tables in a local hard disk or memory, executing a task assigned by the management node, and saving an  
25 execution result in the local hard disk or memory. The management node and the computing node can be either a computer or a processor.

For example, {names of all Chinese people} is a table which includes 1.6 billion names; since the data amount of the table is too large, it is needed to divide the table into some sub-tables, the total number of names in each sub-table is not more than 10  
30 thousand. In this way, sub-tables {{people surnamed Zhang 1}, {people surnamed Zhang 2}, ... {people surnamed Qian 1}, ...} are formed, a table includes 160 thousand

sub-tables. These sub-tables are distributed to 1000 computing nodes; the first 160 sub-tables may be stored at the first computing node, and so on. This process is data distribution. Placing a sub-table on a computing node needs to be completed through network transmission; if a computing node stores 160 sub-tables, and then, it is needed to transmit a data amount of 160 sub-tables to this computing node; the larger the data amount is, the heavier the burden of network transmission is.

Cross join is a standard operation command between two tables, and its function is similar to the multiplication in daily arithmetic, but the cross join is a product (Cartesian product) of two tables. For example, table 1 is (5, 4), table 2 is (3, 2, 6), that is, the table 1 includes two elements, namely 5 and 4, and the table 2 includes three elements, namely 3, 2 and 6. In daily arithmetic,  $5 \times 5 = 25$ , but the result of (5, 4) cross join (3, 2, 6) may be  $(5 \times 3, 4 \times 3, 5 \times 2, 4 \times 2, 5 \times 6, 4 \times 6) = (15, 12, 10, 8, 30, 24)$ , or  $(5+3, 4+3, 5+2, 4+2, 5+6, 4+6) = (8, 7, 7, 6, 11, 10)$ . All results obtained from these examples form a table, and each of the obtained tables includes  $2 \times 3 = 6$  elements.

Suppose that there are two computing nodes, namely computing node N1 and computing node N2; if the elements in the table 1 and the table 2 are taken as sub-tables, then the sub-tables {5}, {3}, {2} are distributed on N1, and the sub-tables {4}, {6} are distributed on N2. When it is needed to perform cross join computation between two tables, namely ({5}, {4}) cross join ({3}, {2}, {6}), the following problem will occur: completing the cross join computation needs to perform computation between {5} and {6}, computation between {4} and {3}, and computation between {4} and {2}, but the two sub-tables are at different computing nodes, so that computation cannot be performed. At present, for performing the cross join computation normally, each computing node is usually enabled to store all the sub-tables in one table, and the sub-tables in another table are evenly distributed on each computing node, for example, both N1 and N2 store {3}, {2}, {6}, N1 stores {5}, and N2 stores {4}; or, both N1 and N2 store {5} and {4}, N1 stores {3}, {2}, and N2 stores {6}; there are other data distributing modes. However, such existing data distributing mode will cause a very large data amount needed to be transmitted to the computing node, so the burden of network transmission is very heavy; especially when the more the computing nodes to which data are distributed are, and the closer the numbers of the sub-tables of two tables are, the more obvious the problem of large data amount is. Currently, there is no technology which can decrease the amount of data transmitted.

## SUMMARY

In view of this, the disclosure is mainly intended to provide a data distributing method and a management node, which can decrease the amount of data transmitted, and lighten the network transmission load.

5 To this end, the technical solutions of the disclosure are implemented as follows.

The disclosure provides a data distributing method, which is applied to a node; the method may include:

according to a received request for cross join, acquiring the number of sub-tables, serial numbers of sub-tables, the number of nodes and serial numbers of nodes, wherein  
10 the sub-tables and the nodes are separately included in table A and table B which need to perform cross join;

setting the serial numbers of the sub-tables included in the table A or the table B in abscissa or ordinate in sequence, to form length or width of a big rectangle separately to generate the big rectangle;

15 dividing the big rectangle into two sub-rectangles according to a construction method, wherein the two sub-rectangles are respectively a first sub-rectangle and a second sub-rectangle;

evenly dividing the first sub-rectangle and the second sub-rectangle separately to form several small rectangles, wherein each small rectangle corresponds to a serial  
20 number of a computing node and corresponds to a serial number of a sub-table; and

distributing the sub-table corresponding to each small rectangle to the computing node corresponding to each small rectangle.

In an embodiment, dividing the big rectangle into two sub-rectangles according to a construction method may include:

25 computing an ideal backup number  $M_a$  or  $M_b$  of subsets of the table A or the table B, wherein the subsets are distributed to all the computing nodes;

according to the  $M_a$  or  $M_b$  obtained by computing, obtaining an evenly division coefficient  $X_{a1} \times X_{b1}$  of the first sub-rectangle and an evenly division coefficient  $X_{a2} \times X_{b2}$  of the second sub-rectangle;

30 according to the obtained  $X_{a1} \times X_{b1}$  and  $X_{a2} \times X_{b2}$ , computing to obtain a division point  $P_a$  of the length or width of the big rectangle; and

dividing the big rectangle into two sub-rectangles according to the  $P_a$  obtained by computing.

In an embodiment, dividing the big rectangle into two sub-rectangles according to the  $P_a$  obtained by computing may include:

- 5        according to the  $P_a$  obtained by computing, dividing the big rectangle into two sub-rectangles at a side which is set with more serial numbers of sub-tables.

In an embodiment, evenly dividing the first sub-rectangle and the second sub-rectangle separately may include:

- 10        according to the obtained  $X_{a1} \times X_{b1}$  and  $X_{a2} \times X_{b2}$ , evenly dividing the first sub-rectangle and the second sub-rectangle separately.

The disclosure also provides a management node; the management node may include:

- 15        an information acquiring unit, which is configured to, according to the received request for cross join, acquire the number of sub-tables, serial numbers of sub-tables, the number of nodes and serial numbers of nodes, wherein the sub-tables and the nodes are separately included in the table A and the table B which need to perform cross join;

      a big rectangle generating unit, which is configured to set the serial numbers of the sub-tables included in the table A or the table B in abscissa or ordinate in sequence, to form length or width of a big rectangle separately to generate the big rectangle;

- 20        a big rectangle division unit, which is configured to divide the big rectangle into two sub-rectangles according to the construction method, wherein the two sub-rectangles are respectively a first sub-rectangle and a second sub-rectangle;

- 25        a sub-rectangle division unit, which is configured to evenly divide the first sub-rectangle and the second sub-rectangle separately to form several small rectangles, wherein each small rectangle corresponds to a computing node and corresponds to a serial number of a sub-table; and

      a distributing unit, which is configured to distribute the sub-table corresponding to each small rectangle to the computing node corresponding to each small rectangle.

In an embodiment, the big rectangle division unit may include:

- 30        an ideal backup number computing subunit, which is configured to compute an ideal backup number  $M_a$  or  $M_b$  of subsets of the table A or the table B, wherein the subsets

are distributed to all the computing nodes;

an evenly division coefficient computing subunit, which is configured to, according to the  $M_a$  or  $M_b$  obtained by computing through the ideal backup number computing subunit, obtain an evenly division coefficient  $X_{a1} \cdot X_{b1}$  of the first sub-rectangle and an evenly division coefficient  $X_{a2} \cdot X_{b2}$  of the second sub-rectangle;

a division point computing subunit, which is configured to, according to the  $X_{a1} \cdot X_{b1}$  and the  $X_{a2} \cdot X_{b2}$  obtained by the evenly division coefficient computing subunit, compute to obtain the division point  $P_a$  of the length or width of the big rectangle; and

a dividing subunit, which is configured to, according to the  $P_a$  obtained by computing through the division point computing subunit, divide the big rectangle into two sub-rectangles.

In an embodiment, the dividing subunit may be configured to, according to the  $P_a$  obtained by computing through the division point computing subunit, divide the big rectangle into two sub-rectangles at the side which is set with more serial numbers of sub-tables.

In an embodiment, the sub-rectangle division unit may be configured to, according to the  $X_{a1} \cdot X_{b1}$  and the  $X_{a2} \cdot X_{b2}$  obtained by the evenly division coefficient computing subunit, evenly divide the first sub-rectangle and the second sub-rectangle separately.

Given the above, the technical solutions of the disclosure include that the number of sub-tables, serial numbers of sub-tables, the number of nodes and serial numbers of nodes are acquired according to a received request for cross join, wherein the sub-tables and the nodes are separately included in table A and table B which need to perform cross join; the serial numbers of the sub-tables included in the table A or the table B are set in abscissa or ordinate in sequence to form length or width of a big rectangle separately to generate the big rectangle; the big rectangle are divided into two sub-rectangles according to the construction method, wherein the two sub-rectangles are respectively a first sub-rectangle and a second sub-rectangle; the first sub-rectangle and the second sub-rectangle are evenly divide separately to form several small rectangles, wherein each small rectangle corresponds to a serial number of a computing node and corresponds to a serial number of a sub-table; and the sub-table corresponding to each small rectangle are distributed to the computing node corresponding to each small rectangle; thus, the number of subsets which are distributed to the computing nodes can be decreased, and the amount of data transmitted is

reduced.

## **BRIEF DESCRIPTION OF THE DRAWINGS**

Fig. 1 is a flowchart of a first embodiment of a data distributing method provided by the disclosure;

5 Fig. 2 is a structural diagram of an embodiment of a management node provided by the disclosure;

Fig. 3 is a sub-table distribution diagram of a second embodiment of a data distributing method provided by the disclosure;

10 Fig. 4 is a sub-table distribution diagram of a third embodiment of a data distributing method provided by the disclosure; and

Fig. 5 is a structural diagram of a data distributing system.

## **DETAILED DESCRIPTION**

A first embodiment of a data distributing method provided by the disclosure is as shown in Fig. 1, the method includes:

15 step 101: according to a received request for cross join, acquiring the number of sub-tables, serial numbers of sub-tables, the number of nodes and serial numbers of nodes, wherein the sub-tables and the nodes are separately included in table A and table B which need to perform cross join;

20 step 102: setting the serial numbers of the sub-tables included in the table A or the table B in abscissa or ordinate in sequence, to form length or width of a big rectangle separately to generate the big rectangle;

step 103: dividing the big rectangle into two sub-rectangles according to a construction method, wherein the two sub-rectangles are respectively the first sub-rectangle and the second sub-rectangle;

25 step 104: evenly dividing the first sub-rectangle and the second sub-rectangle separately to form several small rectangles, wherein each small rectangle corresponds to a serial number of a computing node, and corresponds to a serial number of a sub-table; and

30 Step 105: distributing the sub-tables corresponding to squares included in each small rectangle to the computing node corresponding to each small rectangle.

In an embodiment, dividing the big rectangle into two sub-rectangles according to a construction method may include:

computing the ideal backup number  $M_a$  or  $M_b$  of sub-tables of the table A or the table B, wherein the sub-tables are distributed to all the computing nodes;

5        according to the  $M_a$  or  $M_b$  obtained by computing, obtaining the evenly division coefficient  $X_{a1} \times X_{b1}$  of the first sub-rectangle and the evenly division coefficient  $X_{a2} \times X_{b2}$  of the second sub-rectangle;

         according to the obtained  $X_{a1} \times X_{b1}$  and  $X_{a2} \times X_{b2}$ , computing to obtain the division point  $P_a$  of the length or width of the big rectangle; and

10       dividing the big rectangle into two sub-rectangles according to the  $P_a$  obtained by computing.

In an embodiment, dividing the big rectangle into two sub-rectangles according to the  $P_a$  obtained by computing may include:

15       according to the  $P_a$  obtained by computing, dividing the big rectangle into two sub-rectangles at the side which is set with more serial numbers of sub-tables.

In an embodiment, evenly dividing the first sub-rectangle and the second sub-rectangle separately may include:

         according to the obtained  $X_{a1} \times X_{b1}$  and  $X_{a2} \times X_{b2}$ , evenly dividing the first sub-rectangle and the second sub-rectangle separately.

20       An embodiment of a management node provided by the disclosure is as shown in Fig. 2, the management node includes:

         an information acquiring unit, which is configured to, according to a received request for cross join, acquire the number of sub-tables, serial numbers of sub-tables, the number of nodes and serial numbers of nodes, wherein the sub-tables and the nodes are  
25       separately included in the table A and the table B which need to perform cross join;

         a big rectangle generating unit, which is configured to set the serial numbers of the sub-tables included in the table A or the table B in abscissa or ordinate in sequence, as length or width of a big rectangle separately to generate the big rectangle;

30       a big rectangle division unit, which is configured to divide the big rectangle into two sub-rectangles according to the construction method, wherein the two sub-rectangles are respectively the first sub-rectangle and the second sub-rectangle;

a sub-rectangle division unit, which is configured to evenly divide the first sub-rectangle and the second sub-rectangle separately to form several small rectangles, wherein each small rectangle corresponds to a serial number of a computing node, and corresponds to a serial number of a sub-table; and

- 5 a distributing unit, which is configured to distribute the sub-tables corresponding to squares included in each small rectangle to the computing node corresponding to each small rectangle.

In an embodiment, the big rectangle division unit may include:

- 10 an ideal backup number computing subunit, which is configured to compute an ideal backup number  $M_a$  or  $M_b$  of subsets of the table A or the table B, wherein the subsets are distributed to all the computing nodes;

- an evenly division coefficient computing subunit, which is configured to, according to the  $M_a$  or  $M_b$  obtained by computing through the ideal backup number computing subunit, obtain the evenly division coefficient  $X_{a1} \times X_{b1}$  of the first sub-rectangle and the  
15 evenly division coefficient  $X_{a2} \times X_{b2}$  of the second sub-rectangle;

a division point computing subunit, which is configured to, according to the  $X_{a1} \times X_{b1}$  and the  $X_{a2} \times X_{b2}$  obtained by the evenly division coefficient computing subunit, compute to obtain the division point  $P_a$  of the length or width of the big rectangle; and

- a dividing subunit, which is configured to, according to the  $P_a$  obtained by  
20 computing through the division point computing subunit, divide the big rectangle into two sub-rectangles.

In an embodiment, the dividing subunit is specifically configured to, according to the  $P_a$  obtained by computing through the division point computing subunit, divide the big rectangle into two sub-rectangles at the side set with more serial numbers of sub-tables.

- 25 In an embodiment, the sub-rectangle division unit is specifically configured to, according to the  $X_{a1} \times X_{b1}$  and the  $X_{a2} \times X_{b2}$  obtained by the evenly division coefficient computing subunit, evenly divide the first sub-rectangle and the second sub-rectangle separately.

- A second embodiment of the data distributing method provided by the disclosure is  
30 introduced below in combination with Fig. 3; in this embodiment, the number  $N_n$  of the computing nodes is 3, the serial numbers of the computing nodes may be expressed as Node1, Node2 and Node3; two tables are expressed as table A and table B respectively;

the number  $N_a$  of sub-tables of the table A is 4, and the serial numbers of sub-tables of the table A may be expressed as  $A_1, A_2, A_3$  and  $A_4$ ; the number  $N_b$  of sub-tables of the table B is 4, and the serial numbers of sub-tables of the table B may be expressed as  $B_1, B_2, B_3$  and  $B_4$ ; the method may include the following steps.

5        Step 201 is that the management node acquires, according to a received request for cross join, the number of sub-tables, the serial numbers of sub-tables, the number of nodes and the serial numbers of nodes, wherein the sub-tables and the nodes are separately included in the table A and the table B which need to perform cross join.

10        Step 202 is that the serial numbers of the sub-tables included in the table A or the table B are set in abscissa or ordinate in sequence to form length or width of a big rectangle separately to generate the big rectangle composed of some squares; the number of the squares is a product of the numbers of the sub-tables separately included in the table A and the table B;

15        in this embodiment, as shown in Fig. 3, the serial numbers of the sub-tables included in the table A are set in the abscissa in sequence as the length, and the serial numbers of the sub-tables included in the table B are set in the ordinate in sequence as the width, so as to generate a big rectangle composed of 16 squares.

20        Step 203 is that the big rectangle is divided into two sub-rectangles according to the construction method, wherein the two sub-rectangles are respectively the first sub-rectangle and the second sub-rectangle;

specifically, this step may include the following steps:

step 2031: computing an ideal backup number  $M_a$  of subsets of the table A or the table B, wherein the subsets are distributed to all the computing nodes;

25        here, the total amount of computation of cross join needed to be completed is  $N_a * N_b$ , and the partial amount of computation of cross join that a computing node can complete is  $(M_a * N_a / N_n) * (M_b * N_b / N_n)$ , and then the product of the partial amount of computation of cross join that a computing node can complete times the number of the computing nodes should be greater than or equal to the total amount of computation of cross join needing to be completed, namely,

30        
$$(M_a * N_a / N_n) * (M_b * N_b / N_n) * N_n \geq N_a * N_b, \quad (1)$$

it can be obtained from formula (1) that  $M_a * N_a * M_b * N_b \geq N_n * N_a * N_b$ , when the

amount of data transmitted is small,  $Ma \cdot Na \cdot Mb \cdot Nb = Nn \cdot Na \cdot Nb$ ;

$Ma \cdot Na + Mb \cdot Nb$  is the amount of data transmitted, according to mathematical principles,  $Ma \cdot Na + Mb \cdot Nb \geq 2\sqrt{Ma \cdot Mb \cdot Na \cdot Nb}$ , when  $Ma \cdot Na = Mb \cdot Nb$ , the amount of data transmitted is minimum,

5 thus,  $Ma = \sqrt{Nn \cdot Na / Nb} = \sqrt{3 \cdot 4 / 4} = 1.73$ ;

step 2032: according to the  $Ma$  obtained by computing, obtaining the evenly division coefficient  $Xa1 \cdot Xb1$  of the first sub-rectangle and the evenly division coefficient  $Xa2 \cdot Xb2$  of the second sub-rectangle;

specifically, according to an equation set:  $Xa1 \cdot Xb1 + Xa2 \cdot Xb2 = Nn$  (2)

10  $Xa1 = \text{integer portion of } Ma$  (3)

$Xa2 = Xa1 + 1$  (4)

since the actual backup number of the sub-tables can be only integer, it is needed to perform integer adjustment to the  $Ma$  to obtain an integer; in this embodiment, it is obtained that  $Xa1=1$ ,  $Xa2=2$ ,  $Xb1=1$ , and  $Xb2=1$ , thus  $Xa1 \cdot Xb1$  is 1,  $Xa2 \cdot Xb2$  is 2. In practice,  $Xb1$  and  $Xb2$  may have multiple solutions separately, as long as a solution is selected from two groups of solutions separately;

step 2033: according to the obtained  $Xa1 \cdot Xb1$  and  $Xa2 \cdot Xb2$ , computing to obtain a division point  $Pa$  of the length or width of the big rectangle, wherein the  $Pa$  means dividing the big rectangle at which square from left;

20 specifically, computing to obtain an ideal value of  $Pa$  according to formula (5),

the ideal value of  $Pa = Na \cdot (Xa1 \cdot Xb1) / (Xa1 \cdot Xb1 + Xa2 \cdot Xb2)$  (5)

in this embodiment, the ideal value of  $Pa = 4 \cdot (1) / (1 + 2) = 1.3$ ,

when the ideal value of  $Pa$  is not an integer, the integer portion of the ideal value of  $Pa$  is taken to obtain the  $Pa$ , and the  $Pa$  is 1 in this embodiment;

25 step 2034: dividing the big rectangle into two sub-rectangles according to the  $Pa$  obtained by computing, wherein the two sub-rectangles are the first sub-rectangle and the second sub-rectangle;

in this embodiment, the big rectangle is divided into left and right sub-rectangles at the first square from left.

Step 204 is that according to the obtained  $Xa1 \times Xb1$  and  $Xa2 \times Xb2$ , the first sub-rectangle and the second sub-rectangle are evenly divided separately;

in this embodiment, the first sub-rectangle is evenly divide according to  $1 \times 1$ , the abscissa of the first sub-rectangle is divided into 1 equal part, and the ordinate of the first sub-rectangle is divided into 1 equal part, namely the first sub-rectangle is not evenly divided; the second sub-rectangle is evenly divided according to  $2 \times 1$ , the abscissa of the second sub-rectangle is divided into 1 equal part, and the ordinate of the second sub-rectangle is divided into 2 equal parts, namely the second sub-rectangle is divided into upper and lower rectangles which are the same; the abscissa of the second sub-rectangle may also be divided into 2 equal parts, and the ordinate of the second sub-rectangle is divided into 1 equal part, namely the second sub-rectangle is divided into left and right rectangles which are the same.

Step 205 is that the sub-tables corresponding to the squares included in each small rectangle are distributed to the computing nodes corresponding to each small rectangle.

With reference to Fig. 3, the sub-table A1 of the table A and the sub-tables B1, B2, B3, B4 of the table B are distributed to the Node1; the sub-tables A2, A3, A4 of the table A and the sub-tables B1, B2 of the table B are distributed to the Node2; the sub-tables A2, A3, A4 of the table A and the sub-tables B3, B4 of the table B are distributed to the Node3; thus, the embodiment of the disclosure only needs to transmit the data amount of 15 sub-tables;

however, adopting the existing technology needs to distribute the sub-tables A1, A2, B1, B2, B3, B4 to the Node1, distribute the sub-tables A3, B1, B2, B3, B4 to the Node2, and distribute the sub-tables A4, B1, B2, B3, B4 to the Node3; that is, it is needed to transmit the data amount of 16 sub-tables; it can be known by comparison that the embodiment of the disclosure can decrease the amount of data transmitted.

A third embodiment of the data distributing method provided by the disclosure is introduced below in combination with Fig. 4; in this embodiment, the number  $Nn$  of the computing nodes is 5, the serial numbers of the computing nodes may be expressed as Node1, Node2, Node3, Node4 and Node5; two tables are expressed as table A and table B respectively; the number  $Na$  of sub-tables of the table A is 5, and the serial numbers of sub-tables of the table A may be expressed as A1, A2, A3, A4 and A5; the number  $Nb$  of sub-tables of the table B is 6, and the serial numbers of sub-tables of the table B may be expressed as B1, B2, B3, B4, B5 and B6; the method may include the following steps.

Step 301 is that the management node acquires, according to a received request for cross join, the number of sub-tables, the serial numbers of sub-tables, the number of nodes and the serial numbers of nodes, wherein the sub-tables and the nodes are separately included in the table A and the table B which need to perform cross join.

5 Step 302 is that the serial numbers of the sub-tables included in the table A or the table B are set in abscissa or ordinate in sequence, as length or width of a big rectangle separately to generate the big rectangle composed of some squares; the number of the squares is a product of the numbers of the sub-tables separately included in the table A and the table B; each square corresponds to the serial numbers of a sub-table of the  
10 table A and a sub-table of the table B;

in this embodiment, as shown in Fig. 4, the serial numbers of the sub-tables included in the table A are set in the abscissa in sequence as the length, and the serial numbers of the sub-tables included in the table B are set in the ordinate in sequence as the width, so as to generate a big rectangle composed of 30 squares.

15 Step 303 is that the big rectangle is divided into two sub-rectangles according to the construction method, wherein the two sub-rectangles are respectively the first sub-rectangle and the second sub-rectangle;

specifically, this step may include the following steps:

step 3031: computing an ideal backup number  $M_a$  of subsets of the table A or the  
20 table B, wherein the subsets are distributed to all the computing nodes;

here, the total amount of computation of cross join needed to be completed is  $N_a * N_b$ , and the partial amount of computation of cross join that a computing node can complete is  $(M_a * N_a / N_n) * (M_b * N_b / N_n)$ , and then the product of the partial amount of computation of cross join that a computing node can complete times the number of the  
25 computing nodes should be greater than or equal to the total amount of computation of cross join needing to be completed, namely

$$(M_a * N_a / N_n) * (M_b * N_b / N_n) * N_n \geq N_a * N_b, \quad (1)$$

it can be obtained from formula (1) that  $M_a * N_a * M_b * N_b \geq N_n * N_a * N_b$ , when the amount of data transmitted is small,  $M_a * N_a * M_b * N_b = N_n * N_a * N_b$ ;

30  $M_a * N_a + M_b * N_b$  is the amount of data transmitted, according to mathematical principles,  $M_a * N_a + M_b * N_b \geq 2\sqrt{M_a * M_b * N_a * N_b}$ , when  $M_a * N_a = M_b * N_b$ , the amount of

data transmitted is minimum,

thus,  $Ma = \sqrt{Nn * Na / Nb} = \sqrt{5 * 5 / 6} = 2.04$

step 3032: according to the Ma obtained by computing, obtaining the evenly division coefficient  $Xa1 * Xb1$  of the first sub-rectangle and the evenly division coefficient  $Xa2 * Xb2$  of the second sub-rectangle;

specifically, according to an equation set:  $Xa1 * Xb1 + Xa2 * Xb2 = Nn$  (2)

$$Xa1 = \text{integer portion of } Ma \quad (3)$$

$$Xa2 = Xa1 + 1 \quad (4)$$

in this embodiment, it is obtained that  $Xa1=2$ ,  $Xa2=3$ ,  $Xb1=1$  and  $Xb2=1$ , thus  $Xa1 * Xb1$  is 2, and  $Xa2 * Xb2$  is 3;

step 3033: according to the obtained  $Xa1 * Xb1$  and  $Xa2 * Xb2$ , computing to obtain a division point Pa of length or width of the big rectangle, wherein the Pa means dividing the big rectangle at which square from left;

specifically, computing to obtain an ideal value of Pa according to formula (5),

the ideal value of  $Pa = Na * (Xa1 * Xb1) / (Xa1 * Xb1 + Xa2 * Xb2)$  (5)

in this embodiment, the ideal value of  $Pa = 5 * (2) / (2 + 3) = 2$ ,

when the ideal value of Pa is an integer, the ideal value of Pa is Pa, and the Pa is 2 in this embodiment;

step 3034: dividing the big rectangle into two sub-rectangles according to the Pa obtained by computing, wherein the two sub-rectangles are the first sub-rectangle and the second sub-rectangle;

in this embodiment, the big rectangle is divided into left and right sub-rectangles at the second square from left.

Step 304 is that according to the obtained  $Xa1 * Xb1$  and  $Xa2 * Xb2$ , the first sub-rectangle and the second sub-rectangle are evenly divided separately;

in this embodiment, the first sub-rectangle is evenly divided according to  $2 * 1$ , the abscissa of the first sub-rectangle is divided into 1 equal part, and the ordinate of the first sub-rectangle is divided into 2 equal parts, namely the first sub-rectangle is divided into upper and lower rectangles which are the same; the second sub-rectangle is evenly divided according to  $3 * 1$ , the abscissa of the second sub-rectangle is divided into 1 equal

part, and the ordinate of the second sub-rectangle is divided into 3 equal parts, namely the second sub-rectangle is divided into 3 rectangles which are the same. In practice, the first sub-rectangle may also be evenly divided according to  $2 \times 1$ , the abscissa of the first sub-rectangle may also be divided into 2 equal parts, and the ordinate of the first sub-rectangle may also be divided into 1 equal part, namely the first sub-rectangle is divided into left and right rectangles which are the same; the second sub-rectangle may also be evenly divided according to  $3 \times 1$ , the abscissa of the second sub-rectangle may also be divided into 3 equal parts, and the ordinate of the second sub-rectangle may also be divided into 1 equal part, namely the second sub-rectangle is divided into 3 rectangles which are the same.

Step 305 is that the sub-tables corresponding to the squares included in each small rectangle are distributed to the computing nodes corresponding to each small rectangle.

With reference to Fig. 4, the sub-tables A1, A2 of the table A and the sub-tables B1, B2, B3 of the table B are distributed to the Node1; the sub-tables A1, A2 of the table A and the sub-tables B4, B5, B6 of the table B are distributed to the Node2; the sub-tables A3, A4, A5 of the table A and the sub-tables B1, B2 of the table B are distributed to the Node3; the sub-tables A3, A4, A5 of the table A and the sub-tables B3, B4 of the table B are distributed to the Node4; the sub-tables A3, A4, A5 of the table A and the sub-tables B5, B6 of the table B are distributed to the Node5; thus, the embodiment of the disclosure only needs to transmit the data amount of 25 sub-tables;

however, adopting the existing technology needs to distribute the sub-tables A1, B1, B2, B3, B4, B5, B6 to the Node1, distribute the sub-tables A2, B1, B2, B3, B4, B5, B6 to the Node2, distribute the sub-tables A3, B1, B2, B3, B4, B5, B6 to the Node3, distribute the sub-tables A4, B1, B2, B3, B4, B5, B6 to the Node4, and distribute the sub-tables A5, B1, B2, B3, B4, B5, B6 to the Node5; that is, it is needed to transmit the data amount of 35 sub-tables; it can be known by comparing that the embodiment of the disclosure can decrease the amount of data transmitted. When the more the computing nodes to which data is distributed are, and the closer the numbers of the sub-tables of two tables are, the greater the effect of the disclosure is.

The above are only the embodiments of the disclosure, and not intended to limit the scope of the claims of the disclosure.

## CLAIMS

What is claimed is:

1. A data distributing method, which is applied to a node, comprising:

acquiring the number of sub-tables, serial numbers of sub-tables, the number of  
5 nodes, and serial numbers of nodes according to a received request for cross join,  
wherein the sub-tables and the nodes are separately included in table A and table B  
which need to perform cross join;

setting the serial numbers of the sub-tables included in the table A or the table B in  
abscissa or ordinate in sequence, to form length or width of a big rectangle separately to  
10 generate the big rectangle;

dividing the big rectangle into two sub-rectangles according to a construction  
method, wherein the two sub-rectangles are respectively a first sub-rectangle and a  
second sub-rectangle;

evenly dividing the first sub-rectangle and the second sub-rectangle separately to  
15 form several small rectangles, wherein each small rectangle corresponds to a serial  
number of a computing node and corresponds to a serial number of a sub-table; and

distributing the sub-table corresponding to each small rectangle to the computing  
node corresponding to each small rectangle.

2. The method according to claim 1, wherein dividing the big rectangle into two  
20 sub-rectangles according to a construction method comprises:

computing an ideal backup number  $M_a$  or  $M_b$  of subsets of the table A or the table B,  
wherein the subsets are distributed to all the computing nodes;

according to the  $M_a$  or  $M_b$  obtained by computing, obtaining an evenly division  
coefficient  $X_{a1} \times X_{b1}$  of the first sub-rectangle and an evenly division coefficient  $X_{a2} \times X_{b2}$   
25 of the second sub-rectangle;

according to the obtained  $X_{a1} \times X_{b1}$  and  $X_{a2} \times X_{b2}$ , computing to obtain a division  
point  $P_a$  of the length or width of the big rectangle; and

dividing the big rectangle into two sub-rectangles according to the  $P_a$  obtained by  
computing.

3. The method according to claim 2, wherein dividing the big rectangle into two  
30

sub-rectangles according to the  $P_a$  obtained by computing comprises:

according to the  $P_a$  obtained by computing, dividing the big rectangle into two sub-rectangles at a side which is set with more serial numbers of sub-tables.

4. The method according to claim 2, wherein evenly dividing the first sub-rectangle  
5 and the second sub-rectangle separately comprises:

according to the obtained  $X_{a1} \times X_{b1}$  and  $X_{a2} \times X_{b2}$ , evenly dividing the first sub-rectangle and the second sub-rectangle separately.

5. A management node, comprising:

an information acquiring unit, which is configured to, according to a received request  
10 for cross join, acquire the number of sub-tables, serial numbers of sub-tables, the number of nodes and serial numbers of nodes, wherein the sub-tables and the nodes are separately included in table A and table B which need to perform cross join;

a big rectangle generating unit, which is configured to set the serial numbers of the sub-tables included in the table A or the table B in abscissa or ordinate in sequence, to  
15 form length or width of a big rectangle separately to generate the big rectangle;

a big rectangle division unit, which is configured to divide the big rectangle into two sub-rectangles according to a construction method, wherein the two sub-rectangles are respectively a first sub-rectangle and a second sub-rectangle;

a sub-rectangle division unit, which is configured to evenly divide the first  
20 sub-rectangle and the second sub-rectangle separately to form several small rectangles, wherein each small rectangle corresponds to a computing node and corresponds to a serial number of a sub-table; and

a distributing unit, which is configured to distribute the sub-table corresponding to each small rectangle to the computing node corresponding to each small rectangle.

25 6. The management node according to claim 5, wherein the big rectangle division unit comprises:

an ideal backup number computing subunit, which is configured to compute an ideal backup number  $M_a$  or  $M_b$  of subsets of the table A or the table B, wherein the subsets are distributed to all the computing nodes;

30 an evenly division coefficient computing subunit, which is configured to, according to the  $M_a$  or  $M_b$  obtained by computing through the ideal backup number computing

subunit, obtain an evenly division coefficient  $Xa1 \cdot Xb1$  of the first sub-rectangle and an evenly division coefficient  $Xa2 \cdot Xb2$  of the second sub-rectangle;

5 a division point computing subunit, which is configured to, according to the  $Xa1 \cdot Xb1$  and the  $Xa2 \cdot Xb2$  obtained by the evenly division coefficient computing subunit, compute to obtain a division point  $Pa$  of the length or width of the big rectangle; and

a dividing subunit, which is configured to, according to the  $Pa$  obtained by computing through the division point computing subunit, divide the big rectangle into two sub-rectangles.

10 7. The management node according to claim 6, wherein the dividing subunit is configured to, according to the  $Pa$  obtained by computing through the division point computing subunit, divide the big rectangle into two sub-rectangles at a side which is set with more serial numbers of sub-tables.

15 8. The management node according to claim 6, wherein the sub-rectangle division unit is configured to, according to the  $Xa1 \cdot Xb1$  and the  $Xa2 \cdot Xb2$  obtained by the evenly division coefficient computing subunit, evenly divide the first sub-rectangle and the second sub-rectangle separately.